

```
# Руководство пользователя: tg-payment-control-bot
```

Что это

`tg-payment-control-bot` – Telegram-бот для контроля ежемесячных платежей в группе:

- создает счет участнику за текущий период;
- показывает статус оплаты;
- в тестовом режиме умеет отмечать оплату как выполненную;
- в момент дедлайна запускает проверку и применяет санкцию для должника.

Требования

- Node.js 20+ (рекомендуется актуальный LTS)
- Docker (для локального PostgreSQL)
- Telegram-бот (токен от BotFather)

Быстрый старт

1) Установите зависимости

```
```bash
npm install
```

### ### 2) Поднимите PostgreSQL

```
```bash
docker compose up -d db
```

3) Заполните ` .env`

Обязательные переменные:

- `BOT_TOKEN` – токен Telegram-бота
- `DATABASE_URL` – строка подключения к PostgreSQL

Рекомендуемые/опциональные:

- `PORT` (по умолчанию `3000`)
- `DEFAULT_AMOUNT_CENTS` (по умолчанию `50000`)
- `DEFAULT_TZ` (по умолчанию `Europe/Berlin`)
- `DEFAULT_DUE_DAY` (по умолчанию `5`)
- `DEFAULT_DUE_HOUR` (по умолчанию `18`)
- `ALLOWED_GROUP_CHAT_ID` (если задан, бот работает только в этой группе)
- `WEBHOOK_BASE_URL` (для будущего webhook-сценария)
- `TEST_DUE_IN_MINUTES` (только для теста: дедлайн через N минут)

Пример `DATABASE_URL`:

```
```env
DATABASE_URL=postgresql://postgres:postgres@localhost:5432/tgpay
```

### ### 4) Инициализируйте схему БД

```
```bash
npm run prisma:push
```

5) Запустите бота

```
```bash
npm run dev
```

## Как пользоваться в Telegram

### `/setup` (в группе, обычно админ)

Подключает текущую группу к боту и сохраняет базовые настройки:

- сумма платежа;
- таймзона;
- день/час дедлайна.

### `/pay`

Создает (или находит уже существующий) счет за текущий месяц и показывает кнопки

⋮